

## **REMARKS**

Claims 1-22 are pending in the present application. Claims 1, 2, 10, 11, 12, 20, 21 and 22 were amended to clarify the claims as described below in section II. Support for the additional amendments to claims 1, 11 and 21 is located at least on page 12, line 15 through page 14, line 19 and on page 15, line 23 through page 16, line 18.

Reconsideration of the claims is respectfully requested.

Amendments were made to the specification to update the PTO serial numbers in the specification on page 1, lines 1-25, as requested by the Examiner. No new matter has been added by any of the amendments to the specification.

### **I. Telephone Interview**

Applicants thank Examiner Camquy Truong for the courtesies extended to Applicants' representative during the December 6, 2004 telephone interview. During the interview, Applicants' representative discussed the distinctions between the claims and the prior art of record. The substance of the telephone interview and additional information is summarized in the following remarks.

### **II. 35 U.S.C. § 112, Second Paragraph**

The Office Action rejects claims 1, 2, 10, 11, 12, 20, 21 and 22 under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter, which applicants regard as the invention. This rejection is respectfully traversed.

The Office Action states:

- a. The following terms lack proper antecedent basis:
  - i. The task identification – claims 1, 2, 11, 12, and 21;
- b. The claim language in the following claims is not clearly understood:
  - ii. As to claims 10 (line 23), 20 and 22, it is not clearly understood whether a task refers to the unique task identification or the task identification or the task identification (i.e. if they are the same, applicant is required to use the same term thru out all of the claim languages); line 23, it is not clearly understand what the relationship is between action and events (i.e. each action has a plurality of events or each event has a plurality of actions or they are the same).

Office Action dated September 8, 2004, page 1.

Claims 1, 2, 11, 12 and 21 are amended by replacing “the task identification” with “the unique task identification” to provide antecedent basis for these claims. Claims 10, 20 and 22 are amended by adding the clause “wherein the corresponding action provides a user friendly description of the related events” to clarify the relationship between “action” and “events” in the claims. Support for the amendment to claims 10, 20 and 22 may be found at least on page 13, lines 8-23 of the specification. Therefore, the rejection of claims 1, 2, 10, 11, 12, 20, 21 and 22 under 35 U.S.C. § 112, second paragraph has been overcome.

### **III. 35 U.S.C. § 103, Alleged Obviousness Based on *Lin* and *Daniel***

The Office Action rejects claims 1-2, 4-12 and 14-22 under 35 U.S.C. § 103(a) as being allegedly unpatentable over *Lin et al.* (U.S. Patent 5,949,415), hereinafter referred to as *Lin*, in view of *Daniel et al.* (U.S. Patent 5,321,837), hereinafter referred to as *Daniel*. This rejection is respectfully traversed.

As to independent claims 1, 11 and 21, the Office Action states:

5. As to claims 1, 11 and 21, *Lin* teaches the invention substantially as claimed including: A method for tracking in logging system (col. 2, line 46), the method comprising:

Receiving, at log task manager (program monitor, col. 4, lines 60-61), a request from an application program to assign a unique task identification to a related events identified by application program (col. 4, lines 20-22 and lines 64-65; col. 5, lines 5-7);

Generating, at a log task manager, the unique task identification (col. 9, lines 42-44);

Attaching the unique task identification to a transport mechanism (col. 6, lines 3 and line 19) that passes information between components (col. 4, lines 60-62); and

Combining the task identification with logging information generated by one of the components (col. 5, lines 3-7 and lines 16-17).

6. *Lin* does not teach filtering a plurality of logging information entries based on the task identification. Specifically, *Lin* does not teach the steps of filtering the received data and correlating them for presentation to a user. However, *Daniel* teaches the steps of filtering the received data and correlating them for presenting to a user (col. 2, lines 12-18 and lines 24-27; col. 4, lines 46-47 and lines 50-52; col. 5, lines 10-15; col. 7, lines 6-10).

7. It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of *Lin* and *Daniel* because

Daniel's correlating of received information would improve the tracking system of Lin by providing a more user friendly system with the timely identification of the root cause of problems/issues in the system.

Office Action dated September 8, 2004, page 2-3.

As amended, claim 1, which is representative of the other rejected independent claims 11 and 21 with regard to similarly recited subject matter, reads as follows:

1. A method for tracking tasks in a logging system, the method comprising:  
receiving, at log task manager, a request associated with an application program to assign a unique task identification to a set of related events having a relationship with a task identified by the application program to be tracked, wherein the relationship between the set of related events and the task is established by the application program;  
generating, at a log task manager, the unique task identification;  
attaching the unique task identification to a transport mechanism that passes information between components;  
combining the unique task identification with logging information generated by one or more of the components to correlate logging information entries associated with related events; and  
filtering a plurality of logging information entries based on the unique task identification to produce a set of correlated logging information entries associated with the related events for presentation to a user. (emphasis added)

Neither *Lin* nor *Daniel*, either alone or in combination, teach or suggest the receiving, combining or filtering steps of independent claims 1, 11 and 21. Specifically, *Lin* and *Daniel* do not teach or suggest receiving, at log task manager, a request associated with an application program to assign a unique task identification to a set of related events having a relationship with a task identified by the application program to be tracked, wherein the relationship between the set of related events and the task is established by the application program. Additionally, *Lin* and *Daniel* do not teach or suggest combining the unique task identification with logging information generated by one or more of the components to correlate logging information entries associated with related events. Further, *Lin* and *Daniel* do not teach or suggest filtering a plurality of logging information entries based on the unique task identification to produce a set of correlated logging information entries associated with the related events for presentation to a user.

*Lin* is directed towards a method and apparatus for tracking program usage in a computer system. The method includes the steps of detecting an executable task or

subtasks executing under the operating system, identifying an application program associated with the executable task or subtask, and tracking information relating to execution of the task or subtask without interfering with their execution. *Daniel* is directed towards an event handling mechanism that filters events into categories or groups of events. Once categorized, the invention associates an action or actions with the categorized event. *Daniel* teaches that an action, such as indicating whether or not to log an event, can be associated to the event after the event is categorized. *Lin* and *Daniel* do not teach or suggest the receiving, combining or filtering steps of independent claims 1, 11 and 21.

In claims 1, 11 and 21, an application program establishes the relationship between the set of related events and the task. The application program identifies a task to be tracked and requests that a unique task identification be assigned to a set of related events having a relationship with the task. Since the unique task identification is attached to a transport mechanism that passes information between components and is combined with logging information generated by one or more of the components to correlate logging information entries associated with related events, all of the different logging components, wherever they may be in the network use the same unique task identification when they are logging related events. The logging information entries associated with related events are correlated through the use of the unique task identification. A plurality of logging information entries are filtered based on the unique task identification to produce a set of correlated logging information entries associated with the related events for presentation to a user. *Lin* and *Daniel* do not teach or suggest these features.

The Office Action refers to the following portions of *Lin* in the rejection of the receiving and combining steps of claims 1, 11 and 21:

Local record file 5 may store a variety of information concerning tasks 6 and subtasks 7 running under operating system 2, such as a LaunchID for each task 6 launched by graphical user interface 3 and a total number of active tasks 6 and subtasks 7.

*Lin*, column 4, lines 20-22.

Referring now to FIG. 2, processing begins when program monitor 4 receives a callback message from operating system 2. Program monitor 4 inspects the callback message to determine whether a task is being started or is ending. In a Windows<sup>TM</sup> environment, the callback message will include a unique identifier,

or handle, for the task. If a task is being started, program monitor 4 further determines whether the task is the first task of an application launched by graphical user interface 3 (Step 120). If so, program monitor 4 calls network computer 8 with a request for the current system time from remote timer 11. Program monitor 4 then creates a new application record in local record file 5 (Step 121). The new application record preferably includes a unique LaunchID for the application, assigned by program monitor 4, a StartDateTime based on the retrieved current system time, and a projected EndDateTime for use in the event graphical user interface 3 terminates prematurely, for example, due to a power outage.

Since, in this example, the current task is the first task spawned by the launched application, program monitor 4 also adds a task record to local record file 5. The task record preferably includes a unique TaskID, such as the Windows<sup>TM</sup>-assigned task handle, as well as the LaunchID to associate the task with the newly-launched application.

*Lin*, column 4, line 60 through column 5, line 17. (emphasis added)

These portions of *Lin* only disclose that a local record file contains application and task records for use in monitoring application program usage in a computer. A program monitor creates an application record in a local record file when a first task is detected for the application, which includes a start time and projected end time. A task record is also added to the local record file, which includes a TaskID for the current task and a LaunchID for the application. *Lin* does not teach or suggest receiving, at log task manager, a request associated with an application program to assign a unique task identification to a set of related events having a relationship with a task identified by the application program to be tracked, wherein the relationship between the set of related events and the task is established by the application program, as recited in claims 1, 11 and 21. *Lin* does not teach that an application program identifies a task to be tracked and requests that a unique task identification be assigned to a set of related events having a relationship with the task. Additionally, *Lin* does not teach that the application program establishes the relationship between the set of related events and the task. Further, *Lin* does not teach or suggest combining the unique task identification with logging information generated by one or more of the components to correlate logging information entries associated with related events, as recited in claims 1, 11 and 21. Additionally, *Daniel* does not teach, suggest or mention these features.

Additionally, *Lin* and *Daniel* do not teach or suggest filtering a plurality of logging information entries based on the unique task identification to produce a set of correlated logging information entries associated with the related events for presentation to a user, as recited in claims 1, 11 and 21. As stated in the Office Action, *Lin* does not teach filtering a plurality of logging information entries based on the task identification and correlating them for presentation to the user. *Daniel* is cited for allegedly teaching the filtering step. To the contrary, *Daniel* teaches filtering events into categories or groups of events.

In the rejection of the filtering step of claim 1, the Office Action refers to the following portions of *Daniel*:

The events of an event stream or streams are "filtered" into categories or groups of events. Once categorized, the invention associates an action or actions with the categorized event. The associated action can be logging the event, routing the event to the electronic address of a user, or sending the event to an application program for further processing.

*Daniel*, column 2, lines 12-18.

The filter entries in turn contain a sequence number, a group identifier, and certain selection criteria. The selection criteria includes a collection of element types, values, and operators.

*Daniel*, column 2, lines 24-27.

Group identifier 51 represents the category in which events can be placed. The selection criteria 52 is used to determine the category in which the subject event belongs. It does so, by using the elements of each event. In particular, the selection criteria is used to determine whether an event's element types and the associated values satisfy the test set forth in the selection criteria itself.

*Daniel*, column 4, lines 46-52.

Each action table entry contains a group identifier (group 70) and at least one associated action 71. Associated actions can be routing the event to one or more electronic addresses of specific users, logging the event in a specific log, or sending the event to an application program, as shown in FIG. 7.

*Daniel*, column 5, lines 10-15.

receiving as input said events;  
applying said events to said filter table; and  
categorizing said events into groups of events based on a comparison of characteristics of said events with information contained in said filter table.

*Daniel*, column 7, lines 6-10.

These portions of *Daniel* only teach that events are filtered into categories or groups of events. Once an event is categorized, an action can be associated with the categorized event. Figure 7 shows the action table with actions, such as “LOG(\*NO)”, “LOG(\*YES)” and “LOG(\*NETATR)”. The actions corresponding to logging only indicate whether or not to log an event and may also indicate a specific log for logging an event based on the variable of the “LOG” action. For example with respect to Figure 7 of *Daniel*, when an event, which is in “GROUP2”, “JOES” or “\*DEFAULT”, occurs then the event is logged in the log specified by the variable “\*NETATR”. If the event is in the “BITBUCKET” group, then the event is not logged. To the contrary, claims 1, 11 and 21 recite filtering a plurality of logging information entries based on the unique task identification to produce a set of correlated logging information entries associated with the related events for presentation to a user. *Daniel* does not teach or suggest this feature.

Neither *Lin* nor *Daniel* teach or suggest receiving, at log task manager, a request associated with an application program to assign a unique task identification to a set of related events having a relationship with a task identified by the application program to be tracked, wherein the relationship between the set of related events and the task is established by the application program. Additionally, *Lin* and *Daniel* do not teach or suggest combining the unique task identification with logging information generated by one or more of the components to correlate logging information entries associated with related events. Finally, *Lin* and *Daniel* do not teach or suggest filtering a plurality of logging information entries based on the unique task identification to produce a correlated set of logging information entries associated with the related events for presentation to a user. Therefore, the alleged combination of *Lin* and *Daniel* does not teach or suggest these features, as recited in independent claims 1, 11 and 21.

Thus, neither *Lin* nor *Daniel*, either alone or in combination, teach or suggest receiving, at log task manager, a request associated with an application program to assign a unique task identification to a set of related events having a relationship with a task identified by the application program to be tracked, wherein the relationship between the set of related events and the task is established by the application program; combining the

unique task identification with logging information generated by one or more of the components to correlate logging information entries associated with related events; and filtering a plurality of logging information entries based on the unique task identification to produce a correlated set of logging information entries associated with the related events for presentation to a user, as recited in independent claims 1, 11 and 21. At least by virtue of their dependency on claims 1, 11 and 21, respectively, neither *Lin* nor *Daniel*, either alone or in combination, teach or suggest the features of dependent claims 2, 4-10, 12, 14-20 and 22. Accordingly, Applicants respectfully request withdrawal of the rejection of claims 1-2, 4-12 and 14-22 under 35 U.S.C. § 103(a).

In addition, with regard to claims 10, 20 and 22, neither *Lin* nor *Daniel*, either alone or in combination, teach or suggest mapping a taskID to a corresponding action, wherein the action provides a user-friendly description of the related events. The cited portions of *Lin* only teach that a task record preferably includes a unique TaskID as well as the LaunchID to associate the task with the newly-launched application. Further, *Daniel* does not teach presenting logging information to a user based on the corresponding action as recited in claims 10, 20 and 22. The cited portions of *Daniel* only teach that an event may be logged.

Thus, in addition to being dependent on their respective independent claims, claims 2, 4-10, 12, 14-20 and 22 are also distinguished over the *Lin* and *Daniel* references based on the specific features recited therein.

#### **IV. 35 U.S.C. § 103, Alleged Obviousness Based on *Lin*, *Daniel* and *Wong***

The Office Action rejects claims 3 and 13 under 35 U.S.C. § 103(a) as being allegedly unpatentable over *Lin* in view of *Daniel* and further in view of *Wong* (U.S. Patent 6,185,288 B1). This rejection is respectfully traversed.

Since claims 3 and 13 depend from independent claims 1 and 11, respectively, the same distinctions between *Lin* and *Daniel*, and the invention recited in claims 1 and 11, apply to dependent claims 3 and 13. Specifically, neither *Lin* nor *Daniel*, either alone or in combination, teach or suggest receiving, at log task manager, a request associated with an application program to assign a unique task identification to a set of related events having a relationship with a task identified by the application program to be tracked,



wherein the relationship between the set of related events and the task is established by the application program; combining the unique task identification with logging information generated by one or more of the components to correlate logging information entries associated with related events; and filtering a plurality of logging information entries based on the unique task identification to produce a correlated set of logging information entries associated with the related events for presentation to a user. In addition, *Wong* does not provide for the deficiencies of *Lin* and *Daniel* with regard to independent claims 1 and 11.

*Wong* is directed toward a system and method for multimedia call signaling. A call setup is coordinated between a calling terminal and another terminal for calls, such as voice calls, FAX calls, video calls, and T.120 conference calls, over a data network, such as the Internet, wireline networks, cellular networks, and PCS networks. *Wong* is cited for allegedly teaching the feature of extending the inheritable thread local at the local thread transport. *Wong* does not teach or suggest the features of receiving, at log task manager, a request associated with an application program to assign a unique task identification to a set of related events having a relationship with a task identified by the application program to be tracked, wherein the relationship between the set of related events and the task is established by the application program; combining the unique task identification with logging information generated by one or more of the components to correlate logging information entries associated with related events; and filtering a plurality of logging information entries based on the unique task identification to produce a correlated set of logging information entries associated with the related events for presentation to a user. Thus, any alleged combination of *Wong* with *Lin* and *Daniel* still would not result in the invention recited in claims 1 and 11 from which claims 3 and 13 depend. Accordingly, Applicants respectfully request withdrawal of the rejection of claims 3 and 13 under 35 U.S.C. § 103(a).

Furthermore, one of ordinary skill in the art, presented only with the alleged combination of *Lin* and *Daniel*, and not having a prior knowledge of Applicants' claimed invention, would not be motivated to combine *Lin*, *Daniel*, and *Wong* to arrive at the invention recited in claims 3 and 13. There is no motivation in *Lin* or *Daniel* to coordinate a call setup between a calling terminal and another terminal. *Lin* is directed

toward a method and apparatus for tracking program usage in a computer system. *Daniel* is directed toward an event handling mechanism that filters events into categories or groups of events. There is no need in either of these references for coordinating a call setup between a calling terminal and another terminal, such as taught by *Wong*, and there is no suggestion in either of *Lin* or *Daniel* to include multimedia call signaling. Thus, any alleged motivation to combine *Lin*, *Daniel*, and *Wong* to attempt to arrive at Applicants' claimed invention in claims 3 and 13 would be completely based on a hindsight knowledge of Applicants' invention and thus, is an improper basis for rejecting claims 3 and 13.

In addition, with regard to claims 3 and 13, *Lin*, *Daniel*, and *Wong*, either alone or in combination, do not teach or suggest extending the inheritable thread local at the local thread transport. The Office Action states that *Lin* and *Daniel* do not teach at the local thread transport, extending the inheritable thread local. The Office Action alleges that this features is taught in the following portion of *Wong*:

Each thread or sub-process is associated with the Calling Process using the call reference specified in each call specification in the mail message.

*Wong*, column 9, lines 42-44.

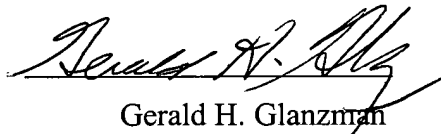
This portion only teaches that each sub-process is associated with the Calling Process, which is a process to handle outgoing calls. Thus, in addition to being dependent on their respective independent claims, claims 3 and 13 are also distinguished over the *Lin*, *Daniel* and *Wong* references based on the specific features recited therein.

V. **Conclusion**

It is respectfully urged that the subject application is patentable over the cited references and is now in condition for allowance. The Examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the Examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

Respectfully submitted,

DATE: January 10, 2005



Gerald H. Glanzman  
Reg. No. 25,035  
Yee & Associates, P.C.  
P.O. Box 802333  
Dallas, TX 75380  
(972) 385-8777  
Attorney for Applicants

GHG/vja